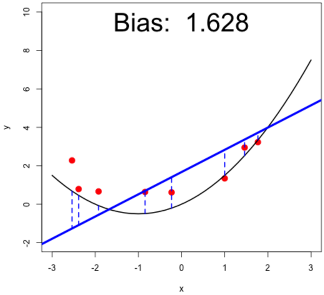
Introduction to Machine Learning

Bias-Variance Decomposition



Learning goals

- Understand how to decompose the generalization error of an inducer into
  - Bias of the inducer
  - Variance of the inducer
  - Noise in the data

BIAS-VARIANCE DECOMPOSITION

Let us take a closer look at the generalization error of a learning algorithm $\mathcal{I}_{L,O}$. This is the expected error an induced model, on trainings sets of size n, when this is applied to a fresh, random test observation.

$$GE_n\left(\mathcal{I}_{L,O}\right) = \mathbb{E}_{\mathcal{D}_n \sim \mathbb{P}_{xy}, (\mathbf{x}, y) \sim \mathbb{P}_{xy}}\left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right) = \mathbb{E}_{\mathcal{D}_n, xy}\left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

We therefore need to take the expectation over all training sets of size n, as well as the independent test observation.

We assume that the data is generated by

$$y = f_{\text{true}}(\mathbf{x}) + \epsilon\,,$$

with normally distributed error $\epsilon \sim \mathcal{N}(0, \sigma^2)$ independent of $\mathbf{x}$.

BIAS-VARIANCE DECOMPOSITION

By plugging in the $L2$ loss $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ we get

$$\begin{aligned} GE_n\left(\mathcal{I}_{L,O}\right) &= \mathbb{E}_{\mathcal{D}_n,xy}\left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right) = \mathbb{E}_{\mathcal{D}_n,xy}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) \\ &= \mathbb{E}_{xy}\underbrace{\left[\mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2 \mid \mathbf{x}, y\right)\right]}_{(*)} \end{aligned}$$

Let us consider the error $(*)$ conditioned on one fixed test observation $(\mathbf{x}, y)$ first. (We omit the $\mid \mathbf{x}, y$ for better readability for now.)

$$\begin{aligned} (*) &= \mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) \\ &= \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y^2\right)}_{=y^2} + \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})^2\right)}_{(1)} - 2\,\underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)}_{(2)} \end{aligned}$$

by using the linearity of the expectation.

BIAS-VARIANCE DECOMPOSITION

$$\mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) = y^2 + \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})^2\right)}_{(1)} - 2\,\underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)}_{(2)} =$$

By using that $\mathbb{E}(z^2) = \text{Var}(z) + \mathbb{E}^2(z)$, we see that

$$= y^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \mathbb{E}^2_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) - 2y\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}))\right)$$

Plug in the definition of y

$$= f_{\text{true}}(\mathbf{x})^2 + 2\epsilon f_{\text{true}}(\mathbf{x}) + \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \mathbb{E}^2_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) - 2(f_{\text{true}}(\mathbf{x}) + \epsilon)\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}))\right)$$

Reorder terms and use the binomial formula

$$= \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 + 2\epsilon\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

BIAS-VARIANCE DECOMPOSITION

$$(*) = \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 + 2\epsilon\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

Let us come back to the generalization error by taking the expectation over all fresh test observations $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$:

$$\begin{aligned} GE_n\left(\mathcal{I}_{L,O}\right) &= \underbrace{\sigma^2}_{\text{Variance of the data}} + \mathbb{E}_{xy}\underbrace{\left[\text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x}, y\right)\right]}_{\text{Variance of inducer at } (\mathbf{x}, y)} \\ &+ \mathbb{E}_{xy}\underbrace{\left[\left(\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 \mid \mathbf{x}, y\right)\right]}_{\text{Squared bias of inducer at } (\mathbf{x}, y)} + \underbrace{0}_{\text{As } \epsilon \text{ is zero-mean and independent}} \end{aligned}$$

BIAS-VARIANCE DECOMPOSITION

$GE_n\left(\mathcal{I}_{L,O}\right) =$

$$\underbrace{\sigma^2}_{\text{Variance of the data}} + \mathbb{E}_{xy}\underbrace{\left[\mathrm{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x}, y\right)\right]}_{\text{Variance of inducer at } (\mathbf{x}, y)} + \mathbb{E}_{xy}\underbrace{\left[\left(\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 \mid \mathbf{x}, y\right)\right]}_{\text{Squared bias of inducer at } (\mathbf{x}, y)}$$

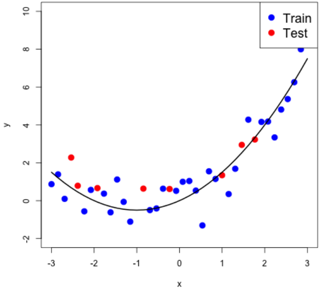
1. The first term expresses the variance of the data. This is **noise** present in the data. Also called Bayes, intrinsic or unavoidable error. No matter what we do, we will never get below this error.
2. The second term expresses how the predictions fluctuate on test-points on average, if we vary the training data. Expresses also the learner's tendency to learn random things irrespective of the real signal (overfitting).
3. The third term says how much we are "off" on average at test locations (underfitting). Models with high capacity have low **bias** and models with low capacity have high **bias**.

BIAS-VARIANCE DECOMPOSITION

Let us consider the following example. We will generate a dataset using the following model :

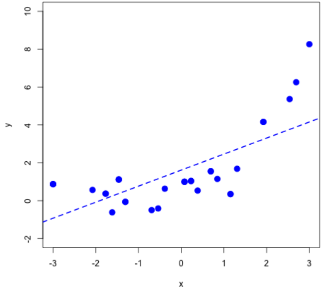
$$y = x + \frac{x^2}{2} + \epsilon \,, \quad \epsilon \sim N(0, 1)$$

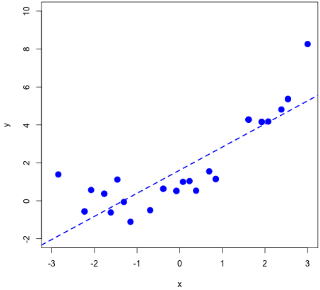
The data is then split in a training set and a test set.



BIAS-VARIANCE DECOMPOSITION

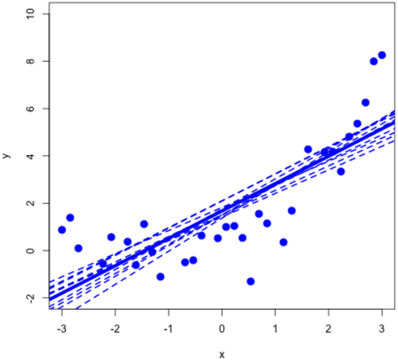
We will train several (low capacity) linear models sampling with replacement from the training dataset. This is commonly known as **bootstrapping**.





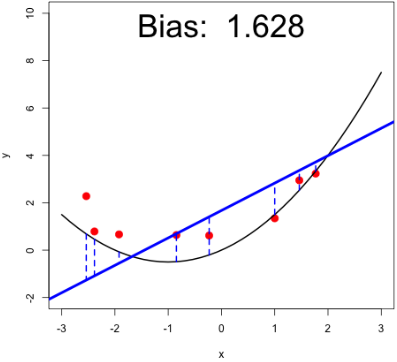
BIAS-VARIANCE DECOMPOSITION

By creating several models, we obtain the average model over different samples of the training dataset.



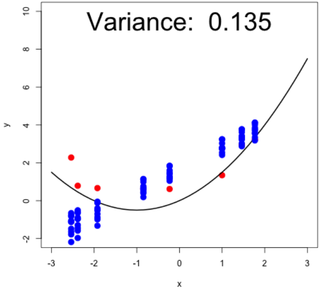
BIAS-VARIANCE DECOMPOSITION

We can now evaluate the squared bias, by computing the average squared difference between the average model and the true model, on the location of the test points.



BIAS-VARIANCE DECOMPOSITION

We may also calculate the average variance of the predictions of the models we trained, at the test points location.

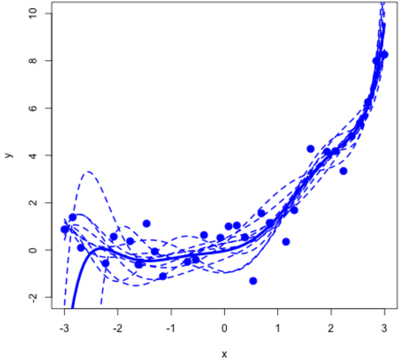


The generalization error is then:

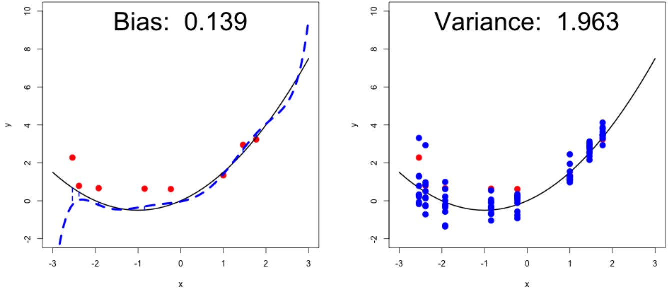
$$GE_n(\mathcal{I}_{L,O}) = 1 + 1.628 + 0.135 = 2.763$$

BIAS-VARIANCE DECOMPOSITION

We will repeat the same procedure, but using a high-degree polynomial that has more capacity.



BIAS-VARIANCE DECOMPOSITION

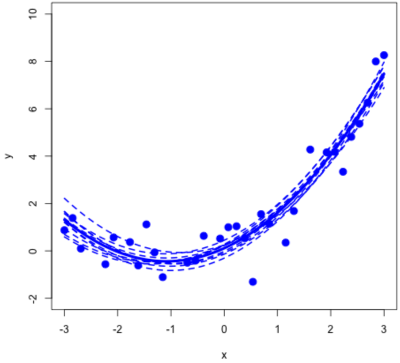


The generalization error is then:

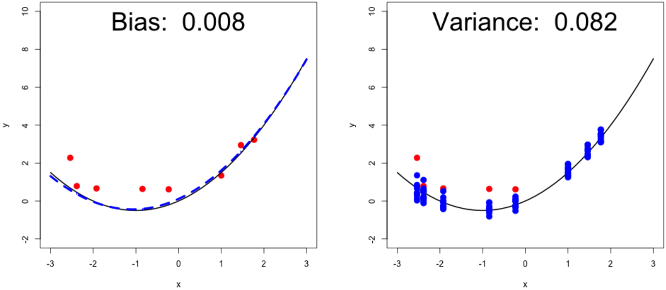
$$GE_n\left(\mathcal{I}_{L,O}\right) = 1 + 0.139 + 1.963 = 3.102$$

BIAS-VARIANCE DECOMPOSITION

What happens if we use a model with the same complexity as the true model?



BIAS-VARIANCE DECOMPOSITION



The generalization error is then:

$$GE_n\left(\mathcal{I}_{L,O}\right) = 1 + 0.008 + 0.082 = 1.091$$

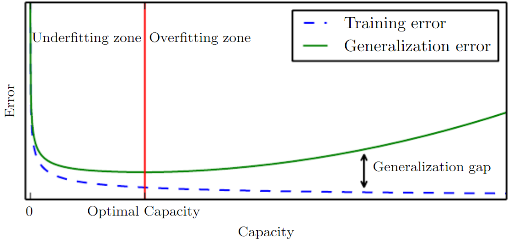
CAPACITY AND OVERFITTING

- The performance of a learner depends on its ability to
  - **fit** the training data well
  - **generalize** to new data
- Failure of the first point is called **underfitting**
- Failure of the second item is called **overfitting**

In our examples we could see that:

- The linear model failed to fit the training data well and thus underfitted (high-bias).
- The high-degree polynomial model failed to generalize to new data and thus overfitted (high-variance).
- The best Generalization error is obtained when the model has the correct complexity.
- Even if the model is correct, there is a lower boundary for the error: The Variance of the data.

CAPACITY AND OVERFITTING



Credit: Ian Goodfellow

- The tendency of a model to over/under fit is a function of its capacity, determined by the type of hypotheses it can learn.
- The generalization error is minimized when it has the right capacity.